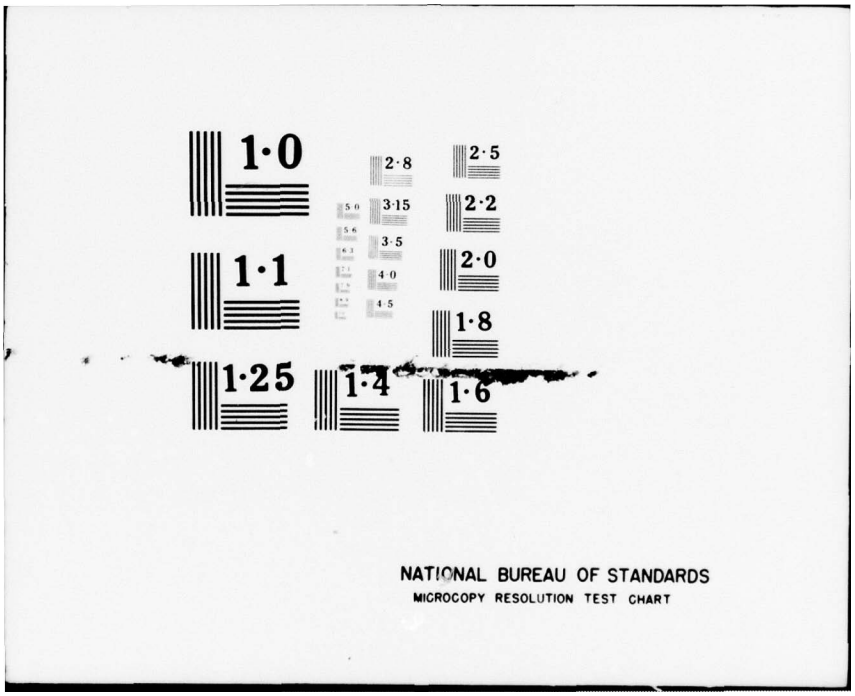
AD-A040 813 NAVAL UNDERWATER SYSTEMS CENTER NEW LONDON CONN NEW --ETC F/G 12/1
CONSTANT-Q SPECTRAL ANALYSIS BY MEANS OF AN APPROXIMATE FAST FO--ETC(U)
APR 77 W F MICHAEL

UNCLASSIFIED NUSC-TR-5585 NL

1 OF 1
ADA
040813

END

DATE
FILMED
7-77


1·0
1·1
1·25
1·4
1·6
1·8
2·0
2·2
2·5
2·8
3·15
3·5
4·0
4·5
5·0
5·6
6·3
NATIONAL BUREAU OF STANDARDS
MICROCOPY RESOLUTION TEST CHART

(12)

Constant-Q Spectral Analysis by Means of an Approximate Fast Fourier Transform Technique

William F. Michael, Jr.
Special Projects Department

7 April 1977

NUSC

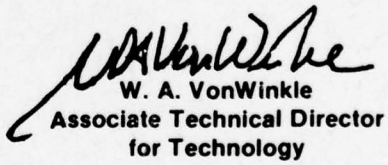
NAVAL UNDERWATER SYSTEMS CENTER
Newport, Rhode Island • New London, Connecticut

PREFACE

This research was performed under NUSC Project No. A-750-11, an Independent Research project, Principal Investigator, H. S. Newman (Code 315). The Navy Program Manager was J. Probus, Naval Material Command (MAT 03521).

The Technical Reviewer for this report was J. J. Wolcin (Code 3153).

REVIEWED AND APPROVED: 7 April 1977



W. A. VonWinkle
Associate Technical Director
for Technology

The author of this report is located at the New London Laboratory, Naval Underwater Systems Center, New London, Connecticut 06320.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: NUSC TR-5585	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): CONSTANT-Q SPECTRAL ANALYSIS BY MEANS OF AN APPROXIMATE FAST FOURIER TRANSFORM TECHNIQUE.	5. TYPE OF REPORT & PERIOD COVERED: Technical Rept.
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): William F. Michael, Jr.	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Naval Underwater Systems Center, New London Laboratory, New London, CT 06320	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: A-750-11
11. CONTROLLING OFFICE NAME AND ADDRESS: Naval Material Command, MAT 03521, Washington, DC 20360	12. REPORT DATE: 7 April 1977
	13. NUMBER OF PAGES: 46
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 76 p.	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Constant-Q Spectral Analysis
FFT
Approximate FFT
Weighted Sum
Overlapped Processing
Side Lobe Control

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A technique for achieving constant-Q spectral analysis by means of an approximate fast Fourier transform (FFT) is presented. Some underlying theory of the approximate FFT and of constant-Q spectral analysis is given. The benefits and drawbacks of this technique are presented. Initial outputs of the technique from a FORTRAN-based Univac 1108 program show that this technique for constant-Q spectral analysis works, and they also illustrate some of the basic characteristics of constant-Q spectral analysis.

DD FORM 1 JAN 73 1473

405918

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF SYMBOLS

a_n	Temporal domain sequence
A_K	Frequency domain sequence
α_i	Expansion factor for $W_i(1)$
B	Control parameter for Kaiser-Bessel function
B_i	Bandwidth of $W_i(1)$
B_o	Normalized bandwidth of W(f)
δ_i	Frequency offset for $W_i(1)$
Δ	Frequency domain sample spacing
Δf	Frequency domain sample spacing of temporal FFT outputs
$\Delta\nu$	Frequency domain sample spacing of vernier FFT outputs
f	Frequency
l_i	FFT bin number on which $W_i(f)$ is centered
L	Temporal window length of an FFT input sequence
π	3.14159265...
R(octave #)	Resolution of uniform resolved spectral estimates in the (octave #) octave.
ν	Resolution of vernier FFT spectral estimates as well as the resolution of uniformly resolved spectral estimates in the first octave.
W(f)	A real, even-frequency domain window function
$W_i(\ell)$	A shifted, expanded, sampled version of W(f)

CONSTANT-Q SPECTRAL ANALYSIS BY MEANS OF AN APPROXIMATE FAST FOURIER TRANSFORM TECHNIQUE

INTRODUCTION

Constant-Q spectral analysis is an important signal processing technique in radar, sonar, speech processing, and other fields where spectral analysis of a signal or an event is desired. There are a plethora of time- and frequency-domain algorithms to achieve constant-Q spectral analysis. This report deals with one such algorithm. Some of the theory behind the technique is covered along with some examples to give the reader a down-to-earth understanding of the technique. A basic knowledge of spectral analysis and fast Fourier transforms (FFT's) is assumed.

The problem of implementation of a constant-Q spectral analysis technique, when real time spectral analysis is desired, lies in the choice of an efficient algorithm from the myriad of available techniques. Efficiency in this sense refers to the joint minimization of computer time and storage while maintaining an acceptable level of accuracy.

This report describes an efficient method of achieving constant-Q spectral analysis. The algorithm results from the merging of two spectral analysis techniques:

1. An approximate FFT technique for octave processing with a different desired resolution for each octave; and

2. A constant-Q spectral analysis technique that is based on weighted sums of uniform-resolution spectral estimates.

As in a human marriage, there are problems in this merging of techniques because of subtleties of the personalities and idiosyncracies of the two partners. These problems are overcome when the idiosyncracies of the partners are understood.

This report is divided into four autonomous, though related, sections:

1. Constant-Q octave processing.

2. The approximate FFT technique for octave processing.

3. The combination of the techniques of 1 and 2 to achieve constant-Q spectral analysis within octaves by means of the approximate

FFT technique.

4. A description and presentation of initial results of a FORTRAN-based algorithm on the Univac 1108 at the Naval Underwater Systems Center (NUSC), New London.

The first section deals with constant-Q spectral analysis. In constant-Q spectral analysis, the resolution of spectral estimates is a fixed percentage of the center frequency of the estimate, and the center frequencies of the spectral estimates are spaced logarithmically. There are a number of ways of achieving constant-Q spectral analysis. The technique chosen in this report uses weighted sums of uniform-resolution spectral estimates to synthesize constant-Q spectral estimates from the uniform spectral estimates. The first section does not delve into the justification of underlying theory of the technique; rather, it covers

1. The selection, sampling, and quantization of the weighting coefficients, and

2. Implementation subtleties of the technique that, if ignored, can lead to serious degradation of the spectral estimates.

The second section describes a technique for performing uniform-resolution spectral estimation within octaves, based on an approximate FFT technique. The approximate FFT technique is an efficient technique that is especially powerful

1. When nonuniform spectral resolution is desired across a frequency band (as in constant-Q spectral analysis);

2. When a zoom-like spectral analysis is desired about a specific frequency or group of frequencies; and

3. When desired spectral resolutions are so narrow that the FFT size would be larger than the memory available.

In the approximate FFT technique, the FFT is treated as a bank of digital-comb filters. Each filter in the comb simultaneously provides the functions of bandpass filtering and complex demodulation of the input data. Just as bandpass filtering is normally done before spectral analysis, so too can the FFT be used as an initial bank of bandpass filters for input to a second set of FFT's which are applied to each filter in the bank. That is, a short-duration FFT is used to separate the input data into coarse spectral bins and a second FFT (a vernier FFT) is applied to successive outputs of each coarse spectral bin to obtain any desired resolution within that bin. Since each bin has its own vernier, there is great latitude in choosing possible types of spectral analysis.

The third section relates the techniques of the first two sections and describes how the two techniques are merged to achieve constant-Q spectral analysis within octaves. The benefits, drawbacks, problems, and solutions to problems that are created by this merger also are covered in the third section.

The fourth section describes the combined spectral-analysis technique as implemented in FORTRAN on the NUSC/NL Univac 1108. The program described is a specialized algorithm, and suggestions for generalizing the technique are given along with some initial outputs of the routine.

CONSTANT-Q SPECTRAL ANALYSIS

Constant-Q spectral analysis is called by many names, for example, proportional bandwidth or constant-percentage spectral analysis. Because of these and other names, information under the heading constant-Q spectral analysis is difficult to find. As mentioned earlier, Constant-Q spectral analysis can be considered a specialized form of conventional spectral analysis, in which the center frequencies of the spectral estimates lie along a logarithmic frequency axis and the bandwidth (resolution) of an estimate is a fixed percentage of the center frequency of that spectral estimate.

Many techniques exist for implementing constant-Q spectral analysis, and the technique presented in this section is an efficient method. Weighted linear combinations of uniform-resolution spectral estimates can be used to generate constant-Q spectral estimates. A recent report by Fred Harris,[1] of the Naval Undersea Center (NUC), shows why the sum approach works.

Implementation of this method involves the following:

1. Breaking an input data sequence into octaves and performing uniform-resolution spectral analysis within each octave; and

2. Applying a set of weighting coefficients (often referred to as filter coefficients) to the uniform spectral estimates to obtain constant-Q spectral estimates.

The first step will be discussed later in this report, and this is where the power and benefits of the approximate FFT technique will become apparent. This section will discuss the second step of this method and will dwell on the following points:

1. How does the weighted-sum method of constant-Q spectral analysis work?

2. What octave processing is, and why the weighted-sum method based on octave processing is efficient.

3. Why there is a need for variable overlapping of the time sequences prior to spectral analysis.

4. The selection, digitization, and quantization of weighting functions for the weighting coefficients.

WEIGHTED SUM METHOD

Assume we have a set of spectral estimates $\left\{A_k\right\}_{k=0}^{\frac{N}{2}-1}$ from a length N digital Fourier transform (DFT) of a real input sequence $\left\{a_n\right\}_{n=0}^{N-1}$. Associated with each A_k is a center frequency, as well as a bandwidth (or resolution) of the estimate. Each estimate has side lobes associated with it. These side lobes are a nuisance, because they represent interference from frequencies outside the frequency of interest. Multiplicative windowing of the input sequence, or convolutional weighting of the unweighted DFT outputs, is generally used to control spurious responses.

Constant-Q spectral analysis by means of weighted sums represents a specialized form of convolutional weighting in the frequency domain. This specialized convolutional form involves the use of a different set of weights for each shift in the convolution. Figures 1 through 3 illustrate the principle involved, while reference 1 gives a brief proof of why these weighted sums represent spectral estimates with arbitary centers and arbitrary bandwidths.

Figures 1 and 2 display spectral window functions W(f) and $W_i(\ell)$. The function W(f) is the Fourier transform of a real, even, temporal-domain weighting function, and $W_i(\ell)$ is a sampled, expanded, frequency-shifted version of W(f). Note that both the frequency offset from an FFT bin and an expansion factor can be specified independently for each set $\{W_i(\ell)\}$. Figure 3 displays the weighting of some undefined DFT outputs with sample spacing Δ, to which the convolutional weighting functions are applied. The center frequency of the resulting estimate is $f_i = \Delta\{\ell_i + \delta_i\}$, while the bandwidth is $B_i = \alpha_i B_o \Delta$. The implications of the relationship of f_i and B_i are important:

1. By proper choice of α_i and Δ, a set of weights can be generated to synthesize proportional-bandwidth spectral estimates from uniform spectral estimates over an octave of frequencies; and

2. The same set of weights used in one octave can be used for all octaves by the proper choice of Δ.

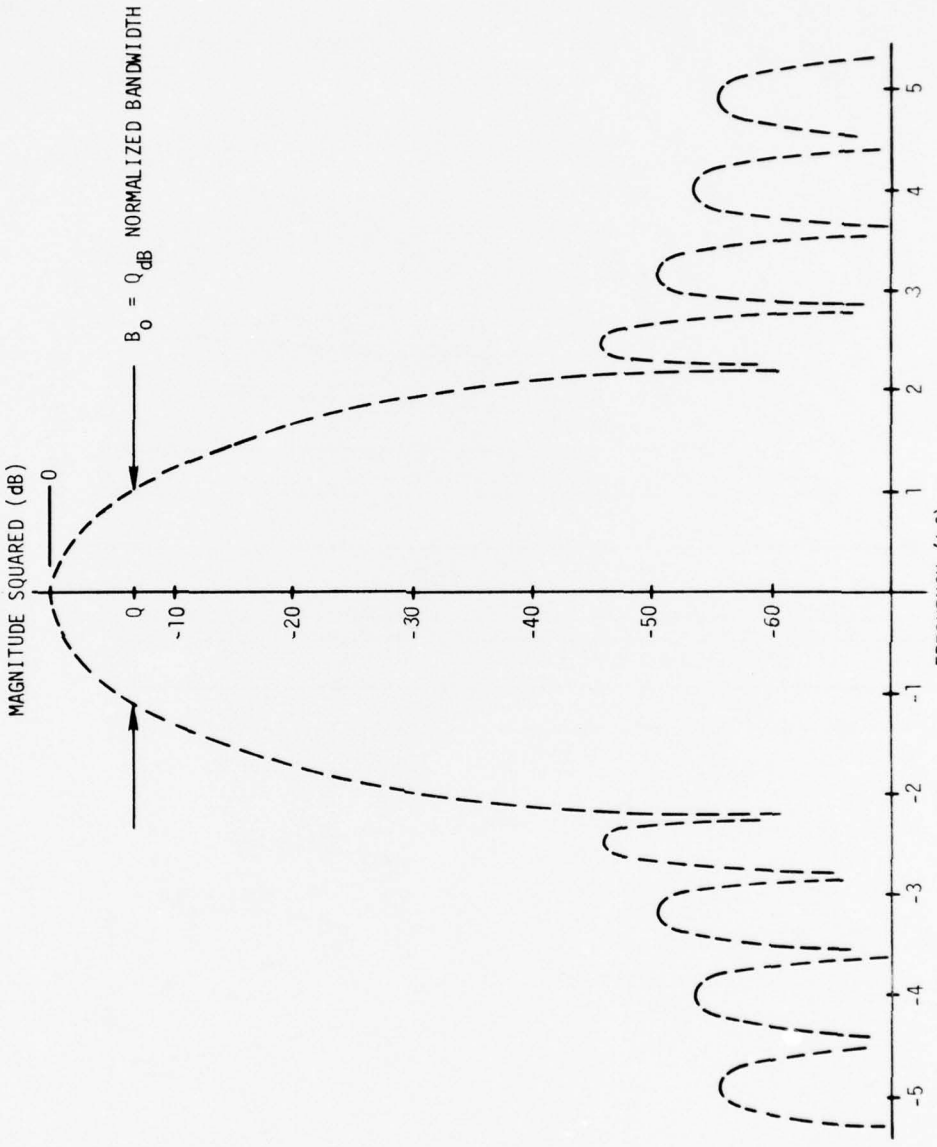


Figure 1. Window Function Sampling, W(f) a Real, Even Window Function

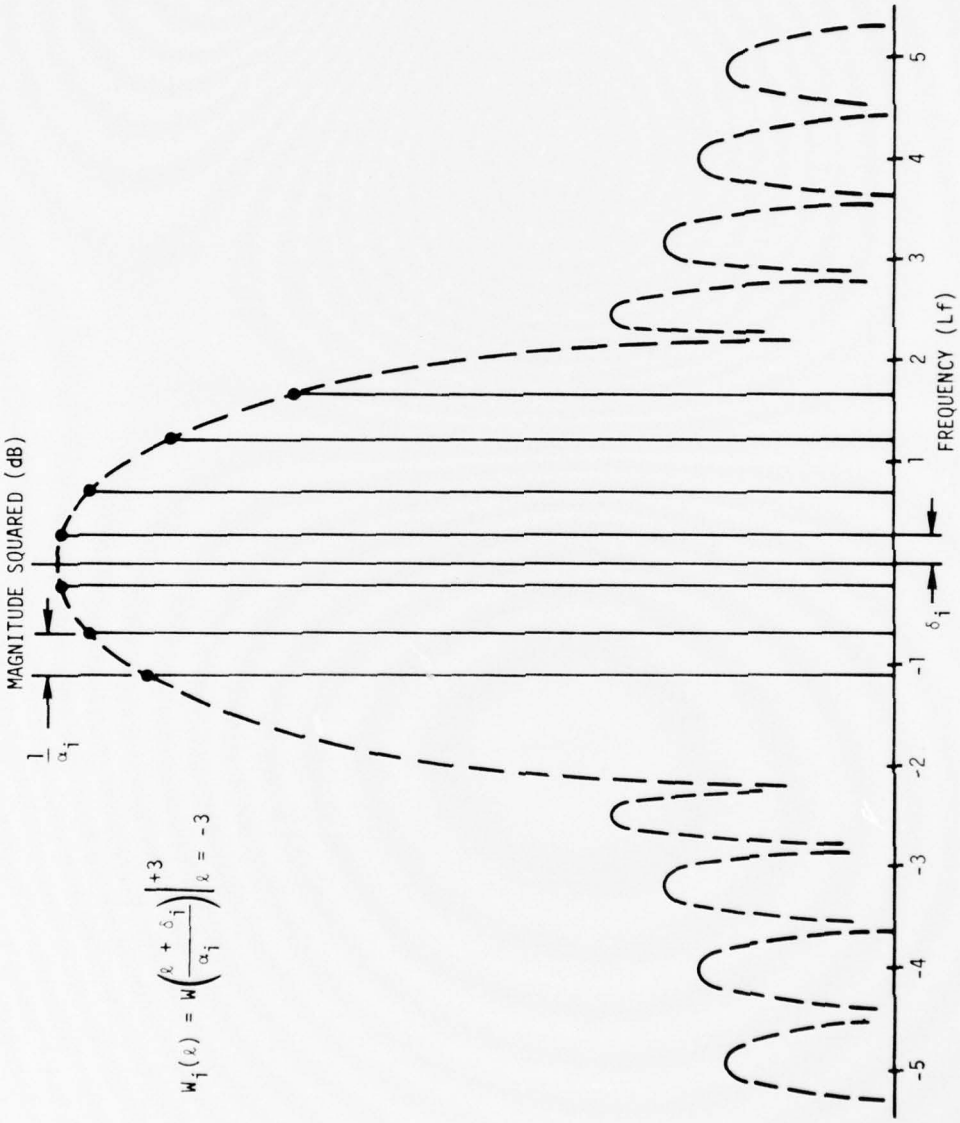


Figure 2. Window Function Sampling, $W_i(\ell)$ a Sampled, Shifted, Expanded W(f)

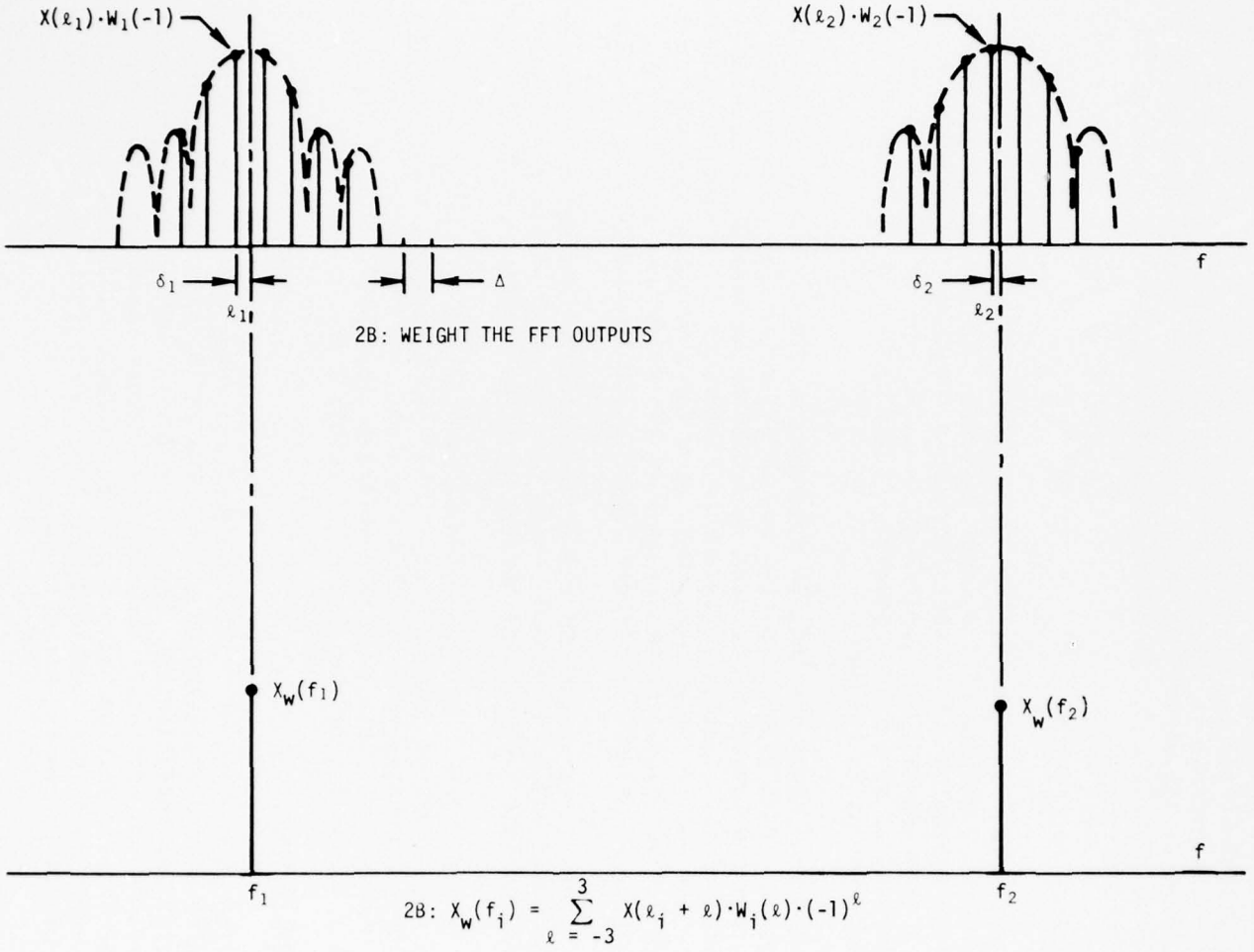


Figure 3. Weighted Sums of Uniform Resolution Spectral Estimates

These two points account for the efficiency of the weighted-sum approach over other techniques. This technique saves computer storage, since only a single set of weights need be stored for a wide band of frequencies because of the octave-processing characteristics of 2, above. The computational savings of the technique lies in the choice of an efficient algorithm for generating the uniform spectral estimates for each octave.

OVERLAPPED PROCESSING

A subtle, often overlooked, point of the weighted-sum technique is the choice of overlap of the input sequences used to create the uniform spectral estimates. With this constant-Q technique, 50 percent-overlapped time segments are sufficient to avoid gaps in the time domain at the low end of an octave. However, at the upper end of an octave a 75 percent overlap is needed.

Figure 4 shows why this is true when contiguous blocks of data are processed. Each filter function has an effective time duration that is less than the length of the data record being processed. Figure 4a illustrates the effective time-domain weightings for a constant-Q filter at the low end of the octave. Note that for contiguous blocks of data the effective weighting creates gaps in the data. The gaps can be covered by 50 percent-overlapped processing (dashed-weighting functions).

Figure 4b shows the effective time-domain weightings for a filter at the upper end of an octave. Note that the time domain gaps are larger and that a 50 percent overlap is insufficient to cover the gaps.

Theoretically, each proportional-bandwidth filter would require a different amount of overlap (ranging between 50 percent and 75 percent overlap), depending upon its location within the octave of interest. This need for variable overlap complicates the application of the filter weights and breaks up the data rates within every octave.

WEIGHTING FUNCTIONS

The generation of filter coefficients falls into the domain of choosing a good weighting function. The choice of a good weighting function was investigated. The Kaiser-Bessel function was found to be convenient and worthwhile and was used in this investigation. The Kaiser-Bessel function has a simply-expressed closed-form Fourier transform and a main lobe width and side lobe levels that can be controlled by a single parameter, B (see figure 5).

Appendix A, written by Dr. A. H. Nuttall, brings out the following points concerning convolutional frequency-domain weights:

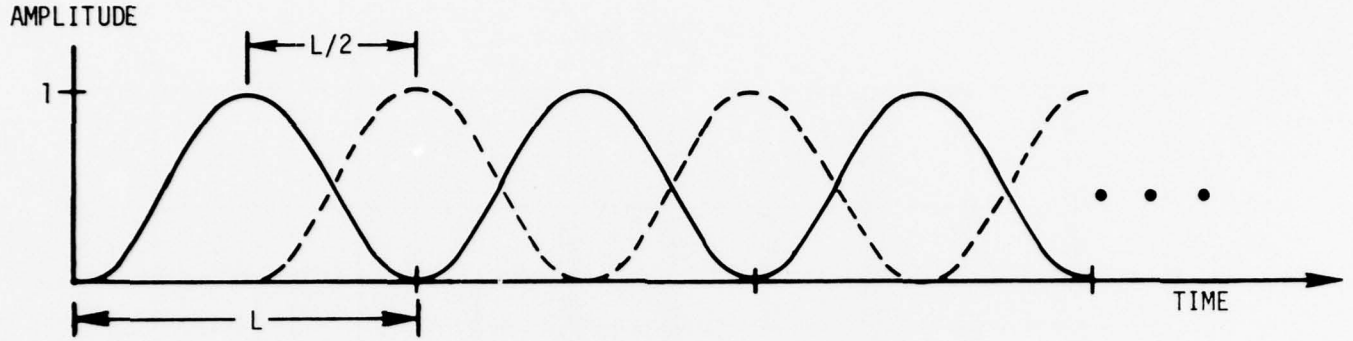


Figure 4a. Lower End of an Octave

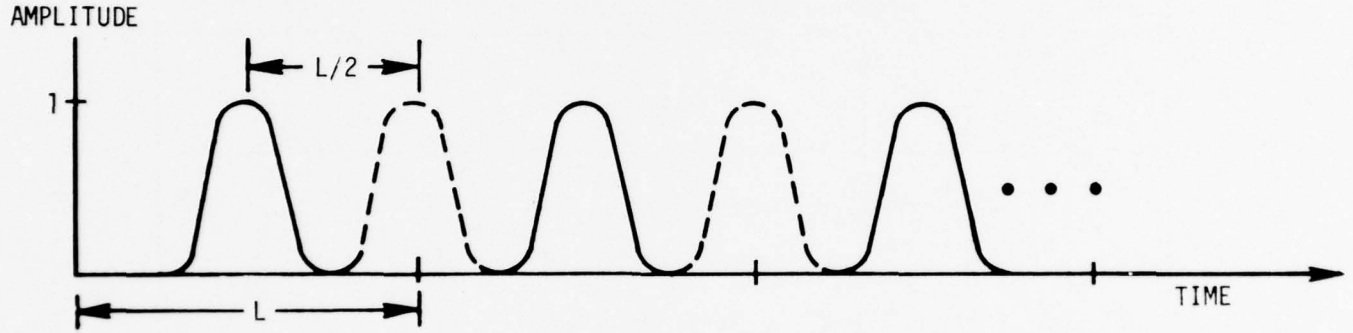


Figure 4b. Upper End of an Octave
Figure 4. Effective Temporal Weightings

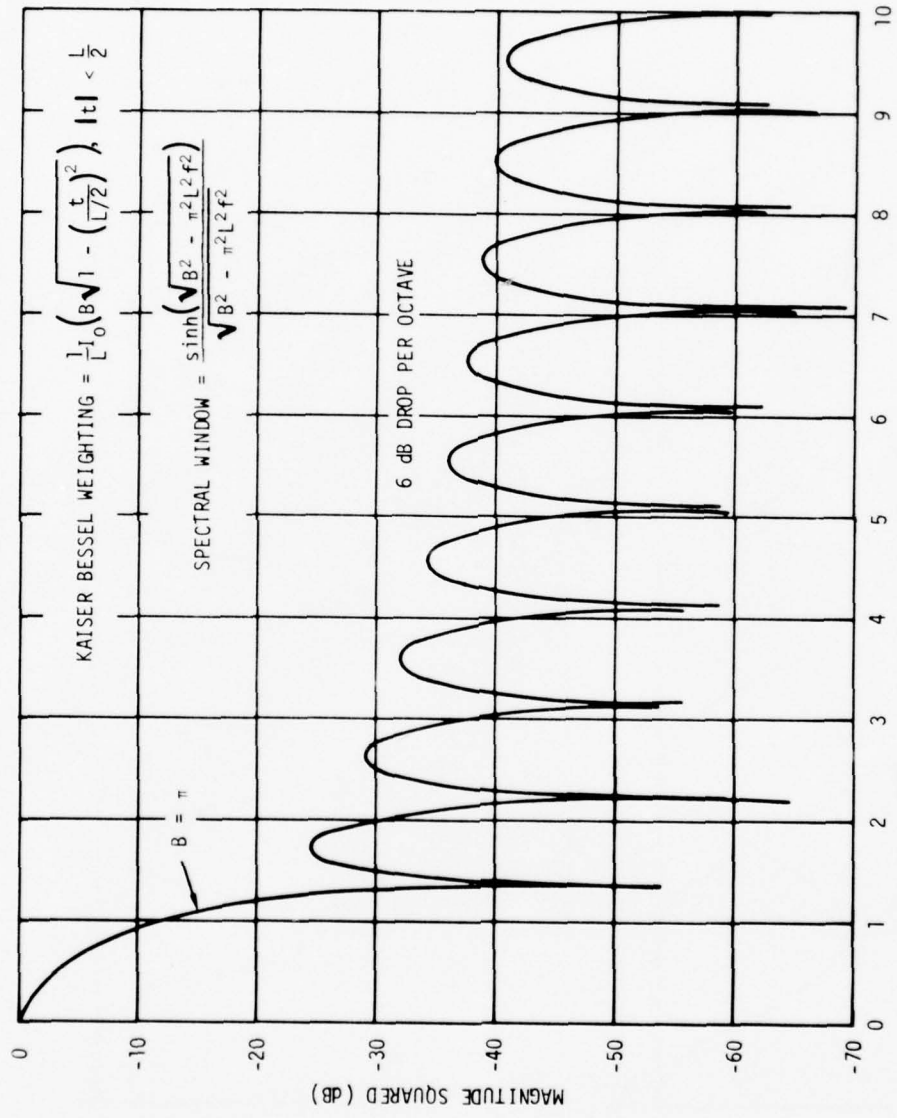


Figure 5. Origin-Centered Kaiser-Bessel Window Function

1. Although the spectral window is, in general, complex when the weighting is not centered about the origin, the spectral window can be represented by real coefficients.

2. Minimum and maximum sampling rates exist when a limited number of samples are used.

3. The width of the main lobe can be changed with negligible effect on side lobe levels by varying the sampling rate within a certain range. For seven coefficients, the maximum range of variation is 2:1.

4. Real filter coefficients can be used to obtain constant-Q spectral estimates from uniformly resolved spectral estimates, but only with the caution mentioned earlier regarding overlapping.

AN APPROXIMATE FFT TECHNIQUE

The approximate FFT technique

1. Accepts input data in moderate-sized time segments, as available;

2. Performs a weighted FFT on each segment (overlapped, if necessary);

3. Stores those frequency portions (at each segment) where narrowband spectral analysis is desired; and

4. Performs a small-size weighted FFT over the total data record available for each frequency portion stored.

The last transform over time (delay) in 4, above, is vernier frequency analysis, as measured from the center of each bin. It is not exact. However, spurious side lobes can be controlled adequately by proper overlapping and weighting in 2 and the use of proper phase factors in 4. By this procedure, we realize narrowband frequency resolution without the need for performing a large-size FFT.

Perhaps the easiest way to present the approximate FFT technique is by a series of simple numerical examples. Suppose we segment the input data into 1 sec sections. In figure 6, a time-frequency diagram illustrates the basic operations required. A temporal weighting (e.g., a Hanning function) is applied to each 1 sec segment of input data, which is then subjected to an FFT. The location along the frequency axis of the resultant spectral components is indicated by a vertical line of X's, spaced 1 Hz apart. For every 50 percent shift (1/2 sec delay) of the temporal weighting, the procedure is repeated eight times, resulting

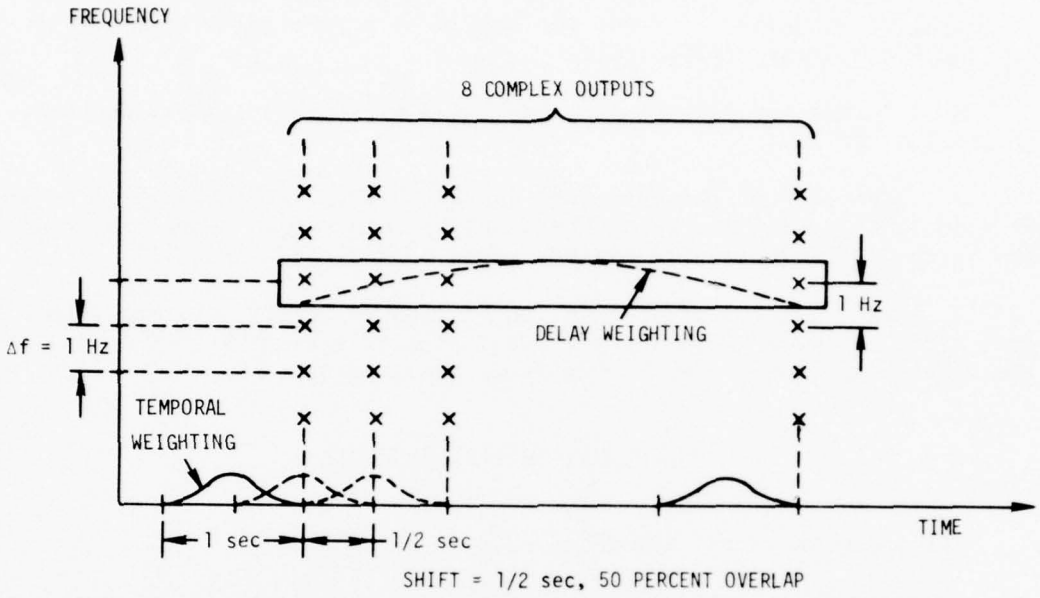


Figure 6. Time-Frequency Diagram

in a series of eight complex spectral outputs covering a time interval of approximately 4 sec. Then the spectral components in a particular frequency bin of interest, as indicated by the horizontal box in figure 6, are (delay) weighted and subjected to a final 8-point (vernier) FFT.

For the example depicting eight outputs in time, each separated by 1/2 sec, the resolution of the vernier FFT is $\Delta\nu$ = 1/4 Hz. Since the total frequency coverage of this vernier FFT is $8\Delta\nu$ = 2 Hz, whereas the original separation of frequency components in figure 6 is only Δf = 1 Hz, only the central half of the vernier outputs about zero frequency are retained. Thus, we can get complete coverage of the frequency scale at $\Delta\nu$ = 1/4 Hz resolution without having to conduct the larger size FFT, i.e., over the entire 4 sec interval.

To indicate in detail how the technique works, consider a unit-amplitude pure tone at 256 Hz with zero phase, as shown in figure 7. To simplify matters, neither temporal weighting nor overlap are employed. The results of the initial 1 sec FFT's are a series of eight ones at a frequency of 256 Hz and zeros at all other frequencies. Then, when the bin centered at 256 Hz is subjected to the 8-point vernier FFT, only the zeroth location (zero frequency) has a nonzero output. Also, vernier FFT's conducted on other bins yield zero outputs everywhere.

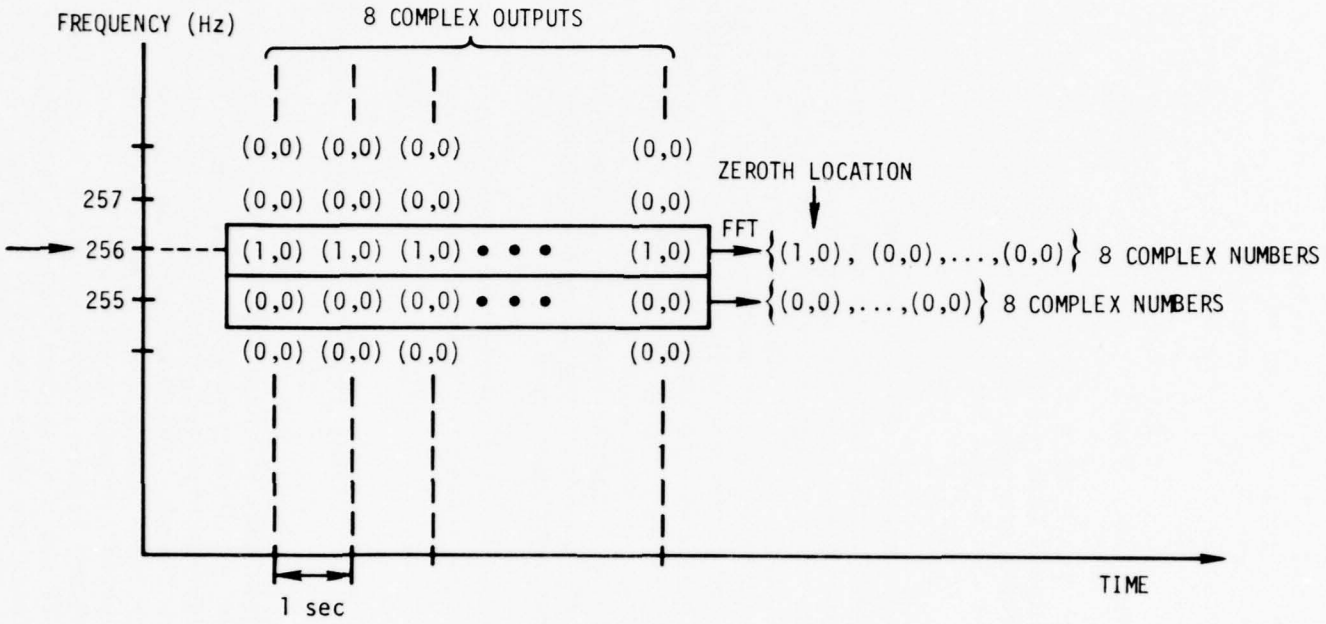


Figure 7. Pure Tone at 256 Hz (Zero Phase) With Unweighted FFT's and No Overlap

In figure 8, a tone is placed instead at 256-1/2 Hz. Now bins centered at 256 and 257 Hz have alternating outputs of magnitude $2/\pi$. The other bins have lesser alternating outputs depending on their proximity to 256-1/2 Hz, and decay according to a sinc-function law for the unweighted example that we are considering. The vernier FFT on the bin centered at 256 Hz now contains only one nonzero element at the fourth location. A similar condition holds for other FFT bins, but with decreased magnitude, depending on their proximity to 256-1/2 Hz.

Finally, in figure 9 a tone is placed at 256-1/8 Hz. The frequency bins have outputs that change by $\exp(i2\pi/8)$ for every 1 sec FFT, and their magnitudes are governed by a sinc-function law. The vernier FFT on the bin at 256 Hz now has only one nonzero output at the first location.

We can see now how the proposed technique works. The location of the peak output of the vernier FFT for a particular frequency bin is a direct measure of the tone frequency present in the input data, relative to the center of that particular bin. In order to control side lobes and spurious responses of this technique, overlap, temporal weighting, delay weighting, and phase factors must be selected and applied carefully. In particular, for 50 percent overlap and Dolph-Chebyshev temporal weighting, -33 dB spurious side lobe responses can be realized. For 75 percent overlap and Dolph-Chebyshev temporal weighting, -86 dB side lobes are attainable. In both cases of overlap, delay weighting is required and was taken as a Hanning function. Although phase factors were unnecessary in the nonoverlap example in figures 7 through 9, they are mandatory when overlap is used.

For a more detailed mathematical treatment of how the technique works and for extensions to other overlaps and weightings, see reference 2.

CONSTANT-Q SPECTRAL ANALYSIS BY MEANS OF THE APPROXIMATE FFT

The approximate FFT technique can now be applied to the weighted-sum method for constant-Q spectral analysis. The weighted-sum method requires uniformly resolved, linearly spaced spectral estimates with a very specific format. The format is based on the octave processing described earlier. Assume for a moment that some arbitrary frequency range has been broken into octaves and the octaves have been numbered, starting with the highest frequency octave as 1, the next lower octave as 2, etc. If one also assumes that the resolution of the spectral estimates within the upper octave is ν, then the required resolution for any octave is $R(\text{octave \#}) = \nu \cdot 2^{-(\text{octave \#} - 1)}$. Another requirement of the specific format is the variable overlap of the time sequences.

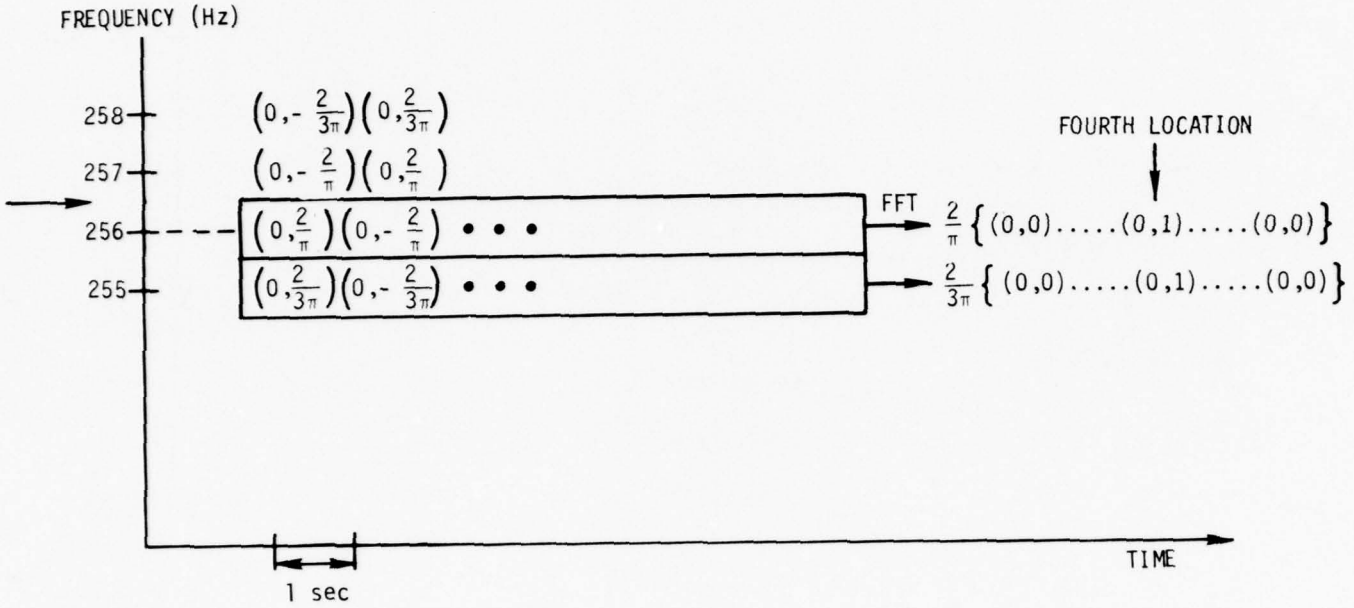


Figure 8. Pure Tone at 256-1/2 Hz With Unweighted FFT's and No Overlap

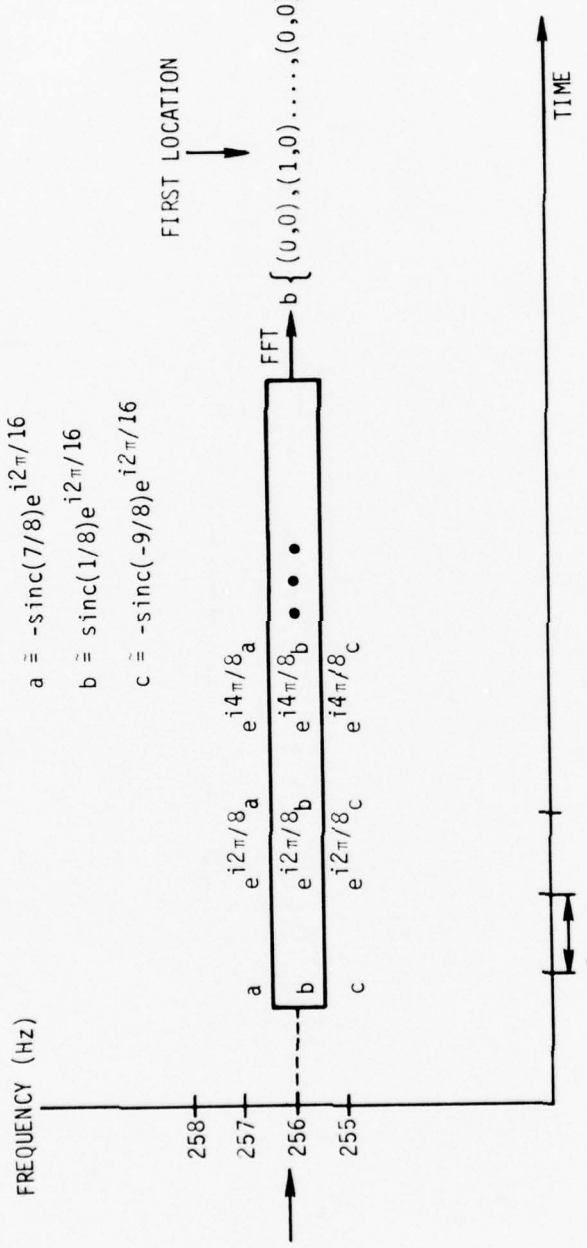


Figure 9. Pure Tone at 256-1/8 Hz With Unweighted FFT's and No Overlap

Figure 10 illustrates the approximate FFT technique implemented as an octave processor. This technique generates with ease the desired pattern of octave resolutions. Instead of selecting single frequencies from each temporal FFT (as in the approximate FFT technique shown in figure 6), we now collect subsets of temporal FFT outputs that represent octaves of interest. For example, one might collect 8 of these subsets for the first octave, 16 for the second octave, etc., and then perform a vernier FFT on the complex output series from each coarse FFT bin within the subset. Each vernier FFT creates uniformly resolved spectral estimates that are then weighted and summed to generate constant-Q spectral estimates. Thus, the proper resolution pattern is achieved through doubling the vernier FFT for each lower octave.

The approximate FFT technique is ideally suited to constant-Q octave processing for the following reasons (figure 11):

1. The temporal FFT allows processing of only the frequencies within the desired octaves;

2. Implementation of the needed resolution within each octave is accomplished by means of the second (vernier) FFT; and

3. A limited variable data overlap can be implemented by overlapping the temporal FFT outputs, if desired.

There are indications that constant-Q octave processing based on the approximate FFT technique offers computational savings over other techniques.

The weighted-sum technique and the approximate FFT technique go together well. The octave processing and variable data overlap required by the weighted-sum approach are well matched to the approximate FFT's characteristics. The benefits of the combined techniques are

1. Computational efficiency;

2. Variability of data overlap; and

3. Elimination of the need for large size FFT's.

There are some difficulties and drawbacks to constant-Q spectral analysis by means of the combined weighted-sum-approximate FFT:

1. Different data rates exist within and between octaves.

2. A picket-fence effect is created by the approximate FFT technique.

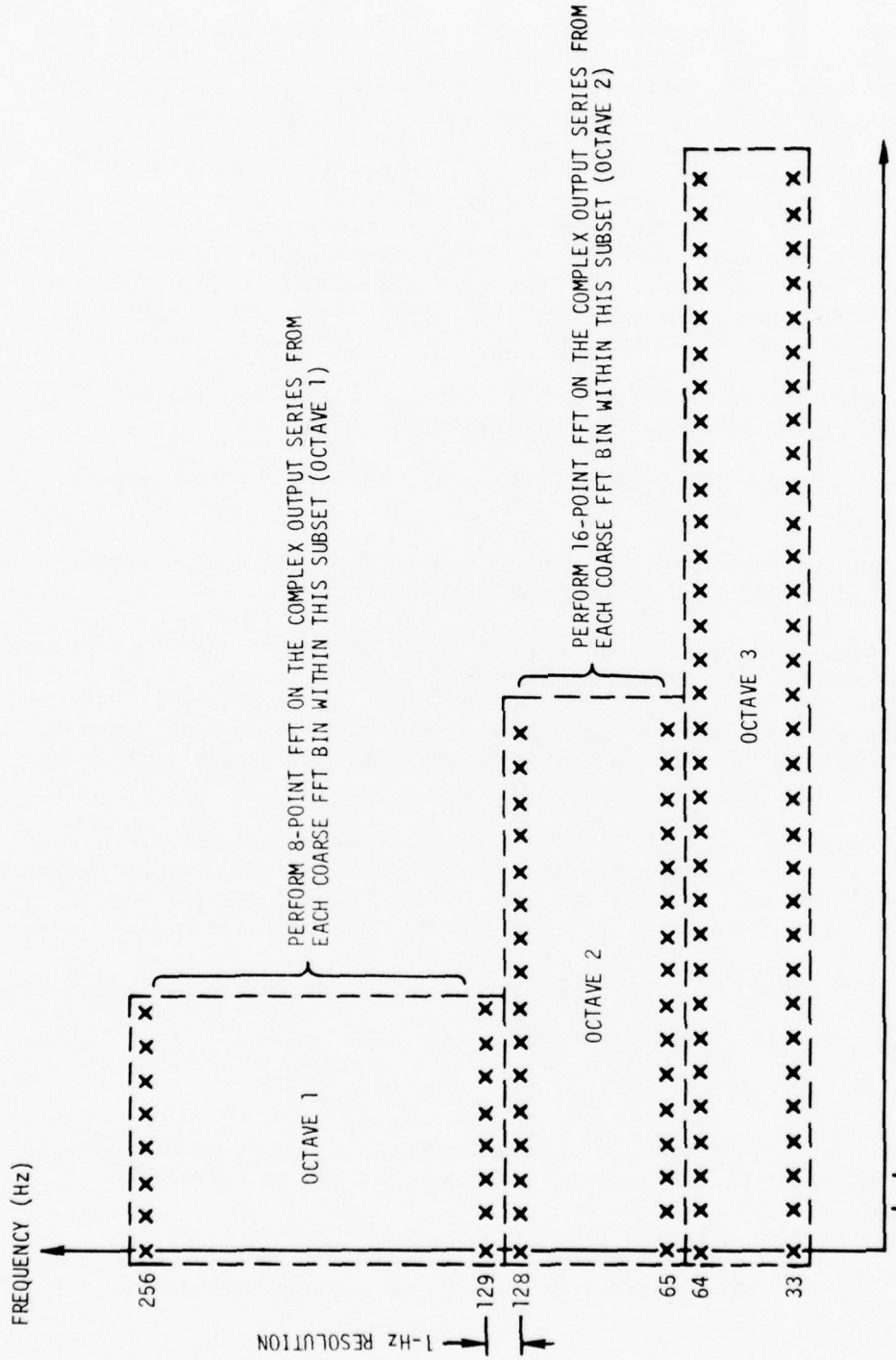


Figure 10. An Octave Processor Implemented by Means of the Approximate FFT Technique

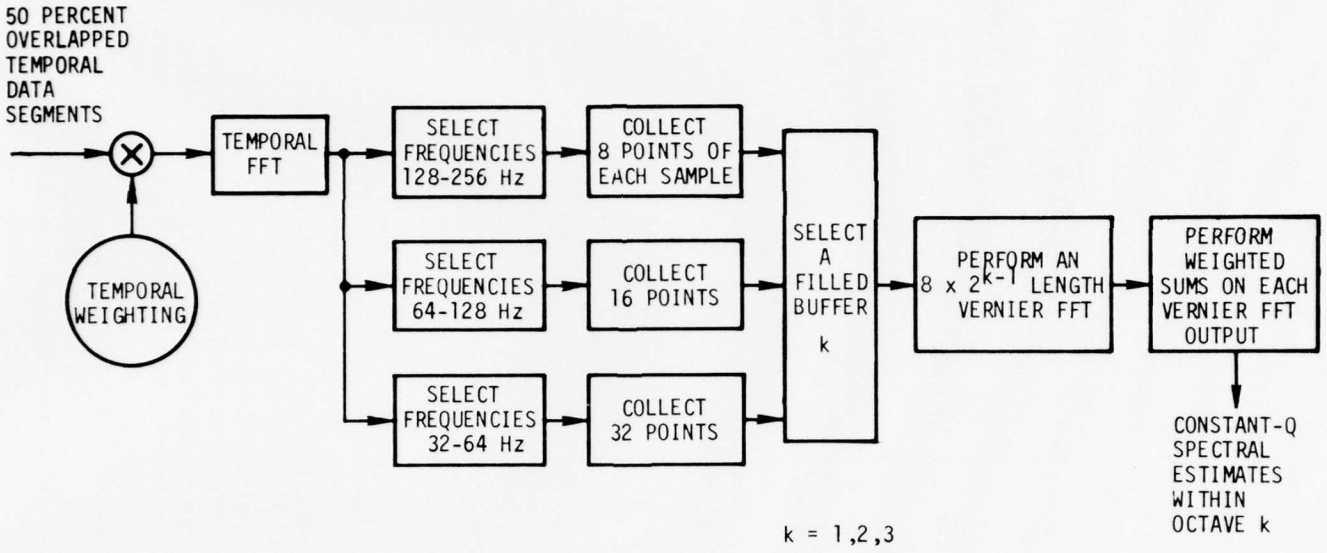


Figure 11. Constant-Q Spectral Analysis by Means of the Approximate FFT Technique

3. The application of the weighting coefficients is complicated slightly.

The existence of variable data rates between octaves is a fundamental characteristic of constant-Q processing. For example, if octave 1 becomes available every second, octave 2 will be available every 2 seconds, octave 3 every 4 seconds, etc. This variable data rate exists regardless of the method used to obtain the constant-Q spectral estimates. When one allows for variable data overlap within an octave, the data rates within an octave also become variable. There is no easy way around this complaint. It is a problem that users of constant-Q spectral analysis techniques learn to accept. An example of the variable data rates can be seen in the next section of this report.

The picket-fence effect of the approximate FFT is caused by the temporal weighting used on the coarse FFT. This effect displays itself when one looks at pure tones centered on vernier FFT outputs. The bin-centered pure-tone outputs will vary in amplitude as the tone varies from bin-center to bin-center within the vernier FFT of a given coarse spectral bin. This variation in amplitude is known and can be corrected for tones centered in vernier FFT bins.

Use of the approximate FFT technique as an octave processor with variable overlap of the temporal FFT outputs complicates the application of the filter weights. Before applying a given filter weight, one must first locate the uniform-resolution bin on which to center the filter weights. In the case of the approximate FFT technique with 50 percent overlap of the temporal FFT's, one must first determine the coarse spectral bin and then the vernier bin on which to center the filter weights.

For example, suppose we had to center a set of weights about f_o = 256-1/16 Hz and had performed a length 8 vernier FFT on coarse spectral bins 255 and 256 Hz. The resulting resolution of the vernier FFT outputs is 1/4 Hz and the desired f_o lies closest to the zeroth vernier bin from the 256 Hz coarse spectral bin's vernier FFT (see figure 12). Figure 12 depicts the center frequencies of vernier FFT outputs rearranged by considering FFT points in the last half of the vernier FFT as negative frequencies. There are redundant spectral estimates, and this is the reason only the central half of the vernier FFT outputs are used.

This example merely illustrates one facet of the implementation of the weighted-sum constant-Q spectral-analysis technique. Generally one not only generates sets of weights representing filters within an octave, but also associates with each set a number that indicates where the filter resides within an octave.

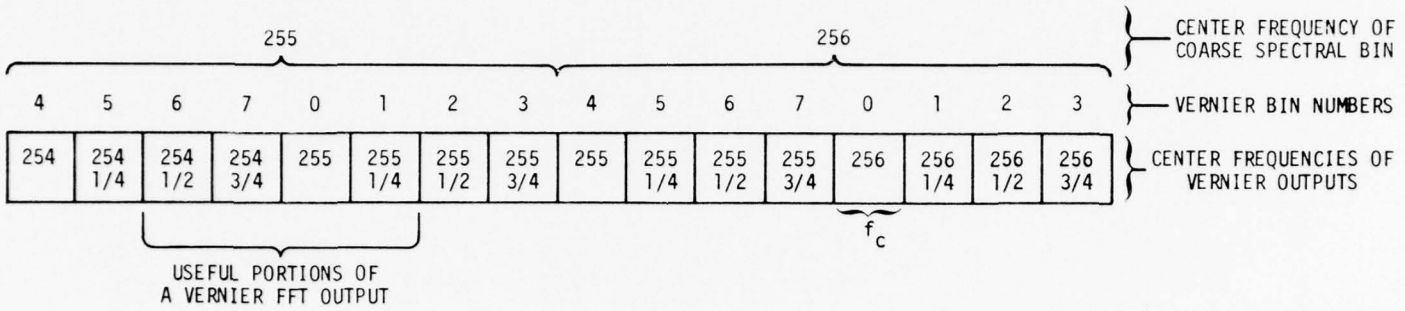


Figure 12. Center Frequencies of Vernier FFT Spectral Estimates

THE UNIVAC 1108 PROGRAM

The FORTRAN-based Univac 1108 program for constant-Q spectral analysis by means of the approximate FFT was originally planned to be a general-purpose check-out routine for the technique. Owing to interfacing problems between existing data generation and display routines, compromises were made that greatly limit the routine's flexibility.

The original specifications for the program were

1. To output either proportional bandwidth or linear spectral estimates;

2. To perform up to four octaves of spectral analysis; and

3. To divide the octaves into a maximum of four sectors, each of which could have a different amount of overlap.

What currently exists is a program that

1. Outputs proportional bandwidth spectral estimates;

2. Performs up to four octaves of spectral analysis; and

3. Divides each octave into two approximately equal halves, where the higher-frequency half has 75 percent overlap while the lower has 50 percent overlap.

Initial outputs of the program are very encouraging and indicate that the merging of the two techniques actually works. What remains to be seen is whether or not the routine is as efficient computationally as originally envisioned.

Figures 13 and 14 are actual normalized outputs from the program. Figure 14 is a compressed version of figure 13 and displays additional time frames.

In the figures, the horizontal axis represents filter numbers (not center frequency). The vertical axis represents time frames. For illustration purposes, the mapping of filter number to frequency and time frame to seconds is irrelevant. Table 1 gives a breakdown of the parameters of the figures.

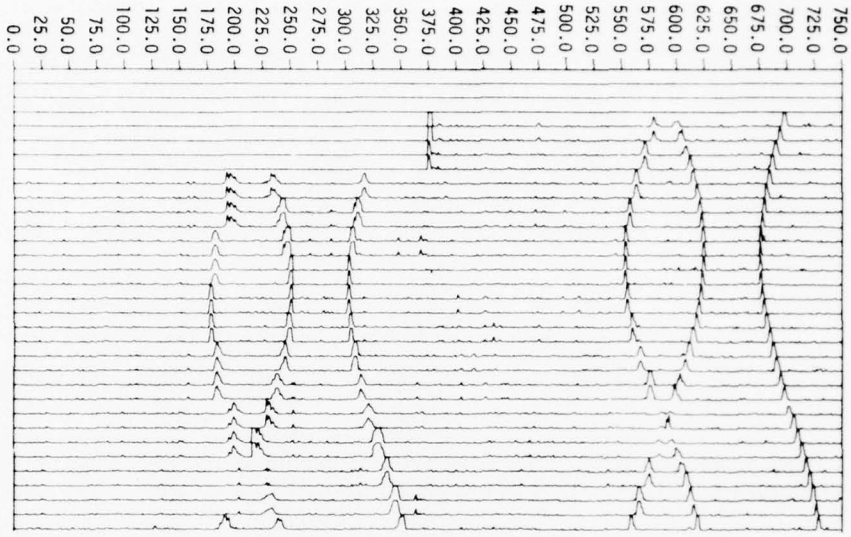


Figure 13. Univac 1108 Constant-Q Outputs, Part I

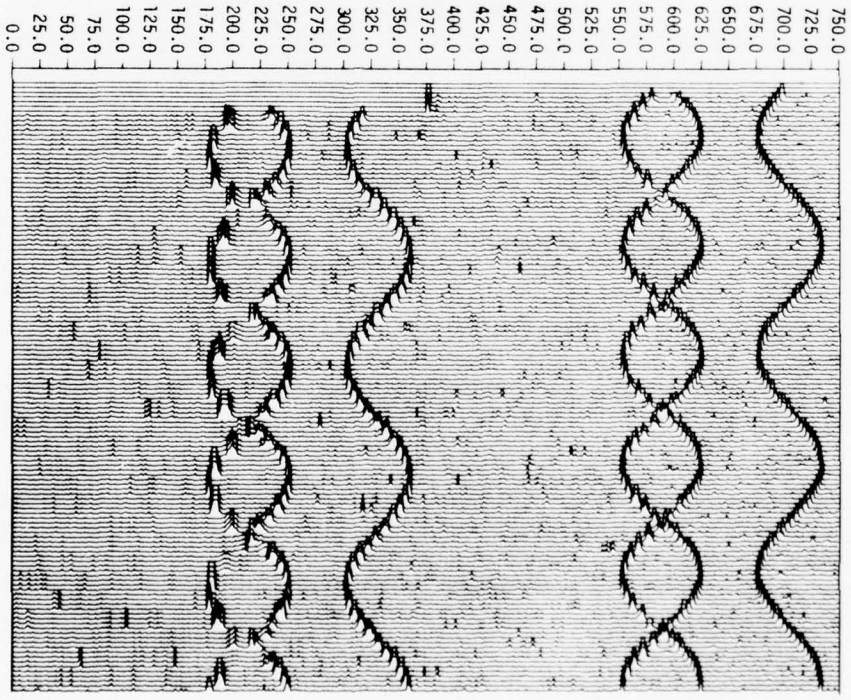


Figure 14. Univac 1108 Constant-Q Outputs, Part II

Table 1. Octave Setups

Octave No.	Sector No.	Filters	Percent Overlap	Output Period (Time Frames/ Output)
1	1	580-750	75	1
1	2	375-580	50	2
2	1	215-375	75	2
2	2	0-215	50	4

The figures illustrate the following characteristics of proportional filtering:

1. The line dynamics are compressed because of the logarithmic frequency spacing of the filter center frequencies; and

2. The varying overlap affects the output data rates of the sectors.

The effects on the data rate can be seen in figure 13. Note that in octave 1, sector 1, each new time frame has a new set of data. In sector 2 of octave 1, the new data set occurs every other time frame. By the time one gets to octave 2, sector 2, the new data set occurs every fourth time frame. The more octaves one processes, the worse this data-output condition gets. Another interesting effect is the startup transient, which also gets worse as one processes more octaves. This startup transient can be seen in the number of time frames needed to obtain nonzero data in a given octave.

The compression of line dynamics can be seen in figure 14. The lines in octave 1 are harmonically related to the similarly varying lines in octave 2 (e.g., the line around filter 715 varies twice as much in frequency as the corresponding line around filter 340). Note that the frequency excursions, in terms of filter numbers, are actually equal.

In order to generalize the program, it would have to be completely rewritten, based upon the basic techniques presented in earlier sections. Things such as data structures are critical to the efficiency of the technique, and perhaps a more efficient method of implementing the overlap (such as phase shifting the filter weights) could be devised.

SUMMARY AND RECOMMENDATIONS

This report has presented a worthwhile and efficient method for obtaining constant-Q spectral analysis, along with some of the pros and cons of the technique. Also represented is a powerful, useful application of the approximate FFT technique.

What lies in the future for the combination of these techniques rests in the hands of persons interested in the method. The author hopes that a valid detailed comparison of this combination of techniques with other techniques, with respect to efficiency and accuracy, will be performed eventually. It would also be instructive if a comparison of proportional-bandwidth spectral analysis with uniform-resolution spectral analysis were performed, to see if the gains in using proportional bandwidth are worth the grief of struggling with variable output rates and other related constant-Q problems.

REFERENCES

1. F. Harris, High Resolution Spectral Analysis with Arbitrary Spectral Centers and Adjustable Spectral Resolutions, NUC Technical Publication 440, February 1975, Naval Undersea Center, San Diego, California.

2. A. H. Nuttall, An Approximate Fast Fourier Transform Technique for Vernier Spectral Analysis, NUSC Technical Report 4767, 16 August 1974, Naval Underwater Systems Center, New London Laboratory.

APPENDIXES

SIDE LOBE CONTROL FOR FREQUENCY DOMAIN SMOOTHING IN PROPORTIONAL BANDWIDTH PROCESSING

Appendixes A and B originally appeared under date of 15 February 1975 as NUSC Technical Memorandum No. TC-4-75, by Albert H. Nuttall.

Appendix A

INTRODUCTION

It is well known that Hanning temporal weighting on a time function is equivalent to frequency domain smoothing of the Discrete Fourier Transform (DFT) coefficients with the sequence -1/4, 1/2, -1/4. However, this fixed smoothing procedure does not allow any variation in the extent (bandwidth) of the frequency domain smoothing effected.

If the time duration of the Hanning temporal weighting were contracted below that of the duration of time over which the DFT were conducted, then a broader frequency domain smoothing would be realized. However, the equivalent frequency domain procedure would be smoothing with a long and more complicated sequence than that given above. It is the purpose of this memorandum to investigate the possibility of using frequency domain smoothing via short sequences for arbitrary smoothing bandwidths and to point out any limitations of the technique.

TECHNICAL CONSIDERATIONS

Let waveform x(t) have voltage density spectrum*

$$X(f) = \int dt\ x(t) \exp(-i2\pi ft). \tag{1}$$

Suppose x(t) is sampled at times $0, \Delta, \ldots, (N-1)\Delta$, where Δ is chosen smaller than $(2f_H)^{-1}$, and f_H is the highest frequency contained in X(f); that is, no aliasing occurs. We compute the DFT of the available samples:

$$A_K \equiv \sum_{n=0}^{N-1} x(n\Delta) \exp(-i2\pi kn/N), \quad 0 \le K \le N-1. \tag{2}$$

But by use of (1), we can express

$$A_K = \sum_{n=0}^{N-1} \exp(-i2\pi kn/N) \int df\ \exp(i2\pi fn\Delta) X(f)$$

$$= \int df\ X(f) \sum_{n=0}^{N-1} \exp\left[-i2\pi n\Delta\left(\frac{k}{N\Delta} - f\right)\right]$$

$$= \int df\ X(f) R\left(\frac{k}{N\Delta} - f\right) = X(f) \circledast R(f)\Big|_{f=\frac{k}{N\Delta}}, \tag{3}$$

*Time functions will be denoted by lower case symbols; their Fourier transforms (frequency functions) will be denoted by capitals.

where

$$R(f) \equiv \sum_{n=0}^{N-1} \exp(-i2\pi n\Delta f)$$

$$= \exp[-i\pi(N-1)\Delta f]\,\frac{\sin(N\pi\Delta f)}{\sin(\pi\Delta f)}. \qquad (4)$$

Equation (3) states that Fourier coefficient A_k is the result of convolving spectrum $X(f)$ with window $R(f)$, and then sampling at frequency $\frac{k}{N\Delta}$. However, $R(f)$ has bad sidelobes ($\sim$-13 dB at $f=\pm\frac{1.5}{N\Delta}$).

In an attempt to reduce sidelobes, suppose we consider the sum of weighted Fourier coefficients (centered about some frequency, f_o, of interest):

$$C(f_o) \equiv \frac{1}{N}\sum_{K=0}^{N-1} A_K B_K(f_o); \qquad (5)$$

weights $\{B_K(f_o)\}$ are yet to be specified. This is the frequency domain smoothing procedure mentioned in the Introduction. Employing (3), (5) becomes

$$C(f_o) = \frac{1}{N}\sum_{K=0}^{N-1} B_K(f_o)\int df\, X(f)\, R\left(\frac{K}{N\Delta} - f\right)$$

$$= \int df\, X(f)\, S^*(f; f_o), \qquad (6)$$

where window function

$$S(f; f_o) \equiv \frac{1}{N}\sum_{k=0}^{N-1} B_k^*(f_o)\, R^*\left(\frac{k}{N\Delta} - f\right). \qquad (7)$$

The problem now is to choose weights $\{B_K(f_o)\}$ such that $S(f; f_o)$ has good sidelobes versus f, in the neighborhood of f_o. If we accomplish this, (6) indicates that coefficient $C(f_o)$ is the result of looking at $X(f)$ through a window $S(f;f_o)$ with good sidelobe behavior.

To get at the choice of weights $\{B_K\}$, consider a real and even weighting function $w_e(t)$ of duration $N\Delta$. Then window $W_e(f)$ is also real and even; see figure 1. Define a time-delayed weighting

$$w(t) = w_e(t-\tau), \qquad (8)$$

Figure 1. Even Weighting and Window

for which the window

$$W(f) = W_e(f) \exp(-i2\pi f\tau); \tag{9}$$

see figure 2. The magnitude characteristics and sidelobes of W(f) are identical

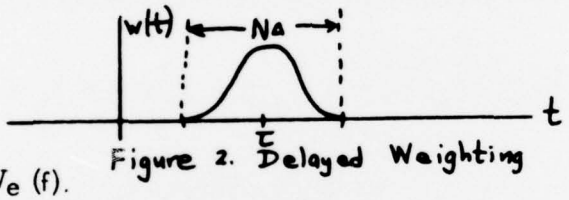


Figure 2. Delayed Weighting

to those of W_e (f).

Now consider samples of a frequency-shifted and expanded version of window W according to

$$W\left(\frac{f-f_o}{\alpha}\right)\Bigg|_{f=\frac{k}{N\Delta}} = W\left(\frac{\frac{k}{N\Delta}-f_o}{\alpha}\right), \tag{10}$$

(where expansion factor α could possibly be a function of shift fo). A plot of $\left|W\left(\frac{f-f_o}{\alpha}\right)\right|$ is given in figure 3. If we employ (9) in (10), we have the samples

Figure 3. Shifted and Expanded Window

$$W_e\left(\frac{\frac{k}{N\Delta}-f_o}{\alpha}\right) \exp\left(-i2\pi \frac{\frac{k}{N\Delta}-f_o}{\alpha}\tau\right). \tag{11}$$

Now if we select

$$\tau = \alpha \frac{N\Delta}{2} I, \tag{12}$$

where I is an integer, (11) can be written as

$$W_e\left(\frac{\frac{k}{N\Delta} - f_o}{\alpha}\right)(-1)^{kI} \exp\left(i\pi f_o N\Delta I\right). \tag{13}$$

That is, the k-dependent terms are real, under the selection (12) for delay τ. The choice of integer I is not yet determined.

We now select the weights $\{B_k(f_o)\}$ to be real, according to the samples

$$B_k(f_o) = \left\{ \begin{array}{l} W_e\left(\frac{\frac{k}{N\Delta} - f_o}{\alpha}\right)(-1)^{kI}, \; 0 \le k_1 \le k \le k_2 \le N-1 \\ \\ 0, \; \text{otherwise} \end{array} \right\}, \tag{14}$$

where

$$\frac{k_1}{N\Delta} < f_o < \frac{k_2}{N\Delta} \tag{15}$$

generally. That is, the mainlobe of window W_e is used for the weights $\{B_k(f_o)\}$. The calculation of weighted Fourier coefficient C(fo) via (5) is thereby simplified by this choice of real weights for $\{B_k(f_o)\}$.

We must now show that the choice (14) results in a window S(f;fo) with good sidelobe properties. Equation (7) yields (using realness of W_e)

$$S(f;f_o) = \frac{1}{N} \sum_{k=k_1}^{k_2} W_e\left(\frac{\frac{k}{N\Delta} - f_o}{\alpha}\right)(-1)^{kI} R^*\left(\frac{k}{N\Delta} - f\right). \tag{16}$$

Now if k_1 and k_2 are chosen such that

$$\left|W_e\left(\frac{\frac{k_{1,2}}{N\Delta} - f_o}{\alpha}\right)\right| << \left|W_e(0)\right|, \tag{17}$$

the edge terms in the sum in (16) are very small compared to the center terms. And if window W_e (f) has a rapidly decaying sidelobe behavior, (16) can be approximated by the infinite sum

$$S(f;f_o) \cong \frac{1}{N}\sum_k W_e\left(\frac{\frac{k}{N\Delta}-f_o}{\alpha}\right)(-1)^{kI}\, R^*\left(\frac{k}{N\Delta}-f\right)$$

$$= \frac{1}{N}\exp(-i\pi f_o N\Delta I)\sum_k W\left(\frac{\frac{k}{N\Delta}-f_o}{\alpha}\right) R^*\left(\frac{k}{N\Delta}-f\right)$$

$$= \frac{1}{N}\exp(-i\pi f_o N\Delta I)\left\{W\left(\frac{f-f_o}{\alpha}\right)\delta_{\frac{1}{N\Delta}}(f)\right\} \circledast R^*(-f), \qquad (18)$$

where impulse train

$$\delta_a(x) \equiv \sum_k \delta(x-ka). \qquad (19)$$

The inverse Fourier transform of S(f;fo) is then available from (18) as

$$s(t;f_o) \equiv \int df\, \exp(i2\pi ft)\, S(f;f_o)$$

$$= \Delta\exp(-i\pi f_o N\Delta I)\left[\delta_{N\Delta}(t) \circledast \left\{\alpha w(\alpha t)\, e^{i2\pi f_o t}\right\}\right] r^*(t)$$

$$= \Delta\exp(-i\pi f_o N\Delta I)\, r^*(t)\sum_k \alpha w\left(\alpha[t-kN\Delta]\right)\exp\left(i2\pi f_o[t-kN\Delta]\right), \qquad (20)$$

where

$$r(t) = \int df\, \exp(i2\pi ft)\, R(f) = \sum_{n=0}^{N-1}\delta(t-n\Delta). \qquad (21)$$

Notice that r(t) is non-zero only in the range $[0,(N-1)\Delta]$.

Now consider the plot of the magnitude of the sum in (20). It is depicted in figures 4A and 4B for I even and I odd, respectively; we have employed

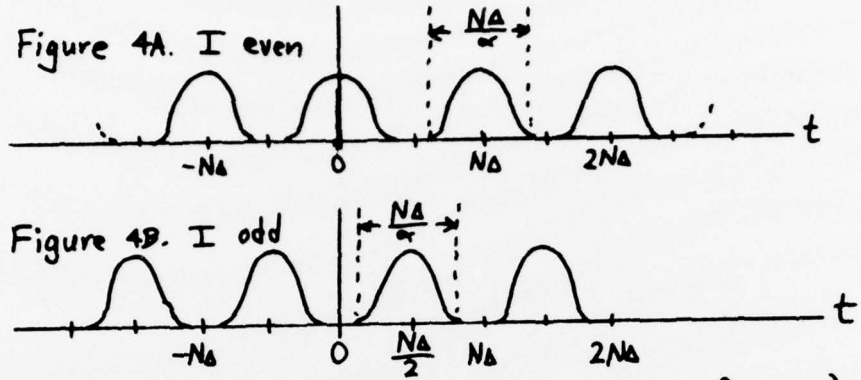


Figure 4. Magnitude of Sum in (20) (drawn for $\alpha > 1$).

figure 2 and (12). The function $r^*(t)$ in (20) gates out the portion $(0, N\Delta)$ in figure 4. If I is even, the resulting function $s(t; f_0)$ has large discontinuities at its edges at 0 and $N\Delta$, thereby leading to a window $S(f; f_0)$ with bad sidelobes; see figure 4A. But if I is odd and if

$$\alpha \geq 1, \tag{22}$$

figure 4B shows that $s(t; f_0)$ will have the smooth behavior dictated by that shown in figure 2. Hence we restrict consideration to I odd in the following.

The only term in (20) which contributes to $s(t; f_0)$ is that for $k = -(I-1)/2$ and yields

$$s(t; f_0) = \Delta \exp\left(-i\pi f_0 N\Delta I\right) r^*(t) \alpha w\left(\alpha\left[t + \tfrac{I-1}{2} N\Delta\right]\right) \cdot \exp\left(i 2\pi f_0 \left[t + \tfrac{I-1}{2} N\Delta\right]\right)$$

$$= \Delta\, r^*(t) \alpha w_e\left(\alpha\left[t - \tfrac{N\Delta}{2}\right]\right) \exp\left(i 2\pi f_0 \left[t - \tfrac{N\Delta}{2}\right]\right), \tag{23}$$

where we have employed (8) and (12). But since, by (22) and figure 1, $w_e\left(\alpha\left[t - \frac{N\Delta}{2}\right]\right)$ is zero outside the interval $(0, N\Delta)$, we can rewrite (23), using (21) and (19), as

$$s(t; f_0) = \Delta\, \delta_\Delta(t) \alpha w_e\left(\alpha\left[t - \tfrac{N\Delta}{2}\right]\right) \exp\left(i 2\pi f_0 \left[t - \tfrac{N\Delta}{2}\right]\right), \tag{24}$$

thereby immediately yielding

$$S(f;f_o) = \delta_{\frac{1}{\Delta}}(f) \circledast \left[W_e\left(\frac{f-f_o}{\alpha}\right) \exp(-i\pi f_o N\Delta) \right]. \qquad (25)$$

The origin impulse in the train $\delta_{\frac{1}{\Delta}}(f)$ yields the term

$$W_e\left(\frac{f-f_o}{\alpha}\right) \exp(-i\pi f N\Delta), \qquad (26)$$

which has the desired magnitude characteristics. The other impulses in $\delta_{\frac{1}{\Delta}}(f)$ lie in frequency ranges where $X(f)$ is zero (by proper anti-aliasing filtering prior to sampling $x(t)$ every Δ seconds). Hence

$$S(f;f_o) \cong W_e\left(\frac{f-f_o}{\alpha}\right) \exp(-i\pi f N\Delta) \qquad (27)$$

in the frequency range of interest. This is the main result of interest.

An alternative expression to (6) for coefficient $C(f_o)$ is available from (1) as

$$\begin{aligned} C(f_o) &= \int df\, S^*(f;f_o) \int dt\, x(t) \exp(-i2\pi f t) \\ &= \int dt\, x(t) \int df\, S^*(f;f_o) \exp(-i2\pi f t) \\ &= \int dt\, x(t)\, s^*(t;f_o). \end{aligned} \qquad (28)$$

But from (24) and figure 1,

$$s^*(t;f_o) = \Delta\, \delta_\Delta(t) \propto w_e\left(\alpha\left[t-\tfrac{N\Delta}{2}\right]\right) \exp\left(-i2\pi f_o\left[t-\tfrac{N\Delta}{2}\right]\right). \qquad (29)$$

The complex exponential in (29) "beats down" the spectrum of $x(t)$ by f_o Hz; the weighting function w_e gates the portion of $x(t)$ indicated in figure 5; and the impulse train δ_Δ samples this gated function. The integral of the product of x and s^* is then taken according to (28). Figure 5 and (28) illustrate that the edges of waveform $x(t)$ (near 0 and $N\Delta$) will receive little or no weight, especially if $\alpha > 1$; this feature may require overlapped processing of $x(t)$.

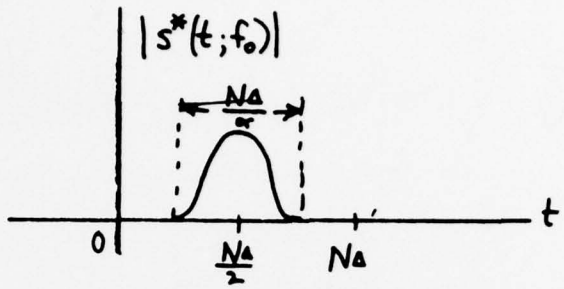


Figure 5. Gating Function

Notice that window $S'(f;f_0)$ in (27) is independent of I (so long as I is odd). Hence we can choose $I=1$ in (14) with no loss in generality. Notice also that the sidelobe characteristics of $S'(f;f_0)$ are exactly those of W_e irrespective of shift f_0 and expansion α, provided that $\alpha \geq 1$ and that (17) is satisfied. Thus we have merely to ensure that we select a good window W_e in the first place when we start computing weights via (14), in order to guarantee a good window $S(f;f_0)$; of course $\alpha \geq 1$ is also necessary.

There is an upper limit on α; that is, arbitrarily large expansion factors in frequency are not allowable. Since K_1 and K_2 are bounded as in (14) and (15), there will come a value of α beyond which the skirts of W_e are not being sampled adequately. The approximation in (18) is then inadequate and the succeeding results inapplicable. For fixed K_1 and K_2, very stringent upper limits on α apply. For example, it is found that for $K_2-K_1=6$ (7 terms in (14)), values of α in the range

$$1 \leq \alpha \leq 2 \tag{30}$$

are allowed for sidelobes in the -30 dB to -40dB range. To go beyond $\alpha=2$ will require more weights than 7. Also additional overlapped processing will be required; recall figure 5.

The restriction to finite K_1 and K_2 in (14)-(16) (rather than the infinite assumption of (18)) makes the sidelobes of $S(f;f_0)$ depend on f_0, rather than the simple dependence in (27). However, these sidelobes can be controlled by a sufficiently large value of K_2-K_1.

Finally, using (27), (6) can be expressed as

$$C(f_0) = \frac{1}{N} \sum_{K=0}^{N-1} A_K B_K(f_0)$$

$$\simeq \int df\, X(f)\, W_e\left(\frac{f-f_0}{\alpha}\right) \exp(i\pi f N\Delta). \qquad (31)$$

The phase factors in (4) and (27) are of no concern as regards their effects in (3) and (6) respectively, since they can be absorbed by a redefinition of the time origin of $x(t)$ in both cases.

DISCUSSION

The desired sidelobe behavior and variable bandwidth expansion factor can be realized through frequency domain smoothing, provided a sufficient number of coefficients are utilized in the frequency domain. However, the larger bandwidth expansion factors will require the use of more coefficients. Additionally, the equivalent time domain operation of multiplication will squelch the edges of the data and may thereby require employment of overlapped processing. These limitations must be kept in mind when this technique is utilized.

Appendix B

PROGRAM

Let $f = f_0 + \frac{r}{N\Delta}$, $0 \le r \le 10$.

(Let $r<0$ in order to look to left of f_0). From (7),

$$S(f; f_0) = \frac{1}{N} \sum_{K=0}^{N-1} B_K^*(f_0)\, R^*\left(-f_0 - \frac{r-K}{N\Delta}\right). \tag{A1}$$

Let

$$f_0 = \frac{M - m_1}{N\Delta}, \quad 0 \le m_1 \le 1, \tag{A2}$$

and

$$K_1 = 0, \quad K_2 = 2M. \tag{A3}$$

Then

$$S(f; f_0) = \frac{1}{N} \sum_{k=0}^{N-1} B_K^*(f_0)\, R^*\left(-\frac{r+M-K-m_1}{N\Delta}\right). \tag{A4}$$

But from (4),

$$\frac{1}{N} R^*\left(\frac{x}{N\Delta}\right) = \exp\left(i\pi \frac{N-1}{N} x\right) \frac{\sin(\pi x)}{N \sin(\pi x/N)} \equiv \ell_N(-x) = \ell_N^*(x). \tag{A5}$$

Therefore

$$S(f; f_0) = \sum_{K=0}^{N-1} B_K^*(f_0)\, \ell_N(r+M-k-m_1) \equiv G(r). \tag{A6}$$

Also

$$B_K(f_0) = (-1)^K W_e\left(\frac{\frac{K}{N\Delta} - f_0}{\alpha}\right) = (-1)^K W_e\left(\frac{k-M+m_1}{\alpha} \frac{1}{N\Delta}\right) \equiv C_{k+1}, \quad 0 \le K \le 2M, \tag{A7}$$

and

$$C_{k+M+1} = (-1)^k W_e\left(\frac{k+m_1}{\alpha} \frac{1}{N\Delta}\right), \quad -M \le k \le M. \text{ (Drop constant } (-1)^M\text{)} \tag{A8}$$

Therefore

$$G(r) = \sum_{K=0}^{2M} C_{k+1}\, \ell_N(r+M-k-m_1)$$

$$= \sum_{k=-M}^{M} C_{k+M+1}\, \ell_N(r-k-m_1). \tag{A9}$$

For the Kaiser-Bessel window,

$$W_e(f) = \frac{\sinh\left(\sqrt{B^2-(\pi N\Delta f)^2}\right)}{\sqrt{B^2-(\pi N\Delta f)^2}}, \qquad (A\,10)$$

$$C_{k+M+1} = (-1)^k \, FNS(x)\Big|_{x=\pi m_2(k+m_1)}, \quad -M \le k \le M, \qquad (A\,11)$$

where

$$FNS(x) \equiv \frac{\sinh\left(\sqrt{B^2-x^2}\right)}{\sqrt{B^2-x^2}}, \qquad (A\,12)$$

$$m_2 \equiv \frac{1}{\alpha}. \qquad (A\,13)$$

For the Hanning window,

$$FNS(x) = \frac{\sin x}{x\left(1-\left(\frac{x}{\pi}\right)^2\right)}. \qquad (A\,14)$$

A basic program for the HP9830A follows.

```
1 REM   NUMBER OF SAMPLES = 2M+1
2 REM   DISPLACEMENT FACTOR IS M1; 0<=M1<=1
3 REM   CONTRACTION FACTOR IS M2; M2<=1
4 REM   PUT -R1 IN 240 & 250 FOR LEFT SIDE
5 REM   WINDOW IN 580
10 M=3
20 B=2*PI
30 N=512
40 DIM C[7]
50 SCALE 0,10,-70,0
60 FOR K=0 TO 10
70 PLOT K,-70
80 PLOT K,0
90 PEN
100 NEXT K
110 FOR K=-70 TO 0 STEP 10
120 PLOT 0,K
130 PLOT 10,K
140 PEN
150 NEXT K
160 M1=0.5
170 M2=1
180 FOR K=-M TO M
190 C[K+M+1]=FNS(PI*M2*(K+M1))*(-1)↑K
200 NEXT K
210 FOR R1=0 TO 10 STEP 0.1
220 G0=G1=0
230 FOR K=-M TO M
240 G0=G0+C[K+M+1]*FNR(R1-K-M1)
250 G1=G1+C[K+M+1]*FNI(R1-K-M1)
260 NEXT K
270 T=G0↑2+G1↑2
280 IF T>0 THEN 310
290 D=-1E+10
300 GOTO 320
310 D=10*LGT(T)
320 IF R1>0 THEN 340
330 O=D
340 D=D-O
350 IF D<-70 THEN 380
360 PLOT R1,D
370 GOTO 390
380 PEN
390 NEXT R1
400 PEN
410 STOP
```

```
420 DEF FNR(R1)
430 IF R1=0 THEN 480
440 R2=PI*R1
450 R3=R2/N
460 R=COS((N-1)*R3)*SIN(R2)/(N*SIN(R3))
470 RETURN R
480 R=1
490 RETURN R
500 DEF FNI(R1)
510 IF R1=0 THEN 560
520 R2=PI*R1
530 R3=R2/N
540 I=SIN((N-1)*R3)*SIN(R2)/(N*SIN(R3))
550 RETURN I
560 I=0
570 RETURN I
580 DEF FNS(X)
590 REM  KAISER BESSEL: SINH(Q)/Q WHERE Q=SQR(B↑2-X↑2)
600 REM  HANNING: SIN(X)/(X*(1-(X/PI)↑2))
610 S1=B↑2-X↑2
620 IF S1 >= 0 THEN 660
630 S1=SQR(-S1)
640 S=SIN(S1)/S1
650 RETURN S
660 IF S1>0 THEN 690
670 S=1
680 RETURN S
690 S1=SQR(S1)
700 S2=EXP(S1)
710 S=(S2-1/S2)/(2*S1)
720 RETURN S
730 END
```

INITIAL DISTRIBUTION LIST

Addressee	No. of Copies
COMOPTEVFOR	1
COMCRUDESGRU2/DESDEVGRU	1
COMSUBDEVGRUONE	1
COMSUBDEVGRUTWO	1
ASN (R&D)	1
ONR (Code 412-3, -8, 480)	3
CNM (Code MAT-03, -03L, -03L4, -03521 (T. A. Kleback), SP-20, ASW-14)	6
DDR&E	2
NAVSHIPRANDCEN, Card (Library)	1
NRL (W. Morrough, R. Roges)	2
NAVOCEANO (Code 7200)	1
NAVAIRSYSCOMHQ (T. Golart)	1
NAVELECSYSCOMHQ (Code PME-124)	1
NAVSEASYSCOMHQ (Code SEA-037, -06H1, -06H1-1, -06H1-2, -06H1-3, -06H1-4, -06H2 (D. Early), -09G3 (4), -660, -661)	13
NAVAIRDEVCEN (T. Castaldi, Mr. Russo)	2
NAVCOASTSYSLAB	1
NAVSURFWPNCEN (M. Stribling)	1
NELC	1
NAVUSEACEN (R. Bolam, P. Moose, H. Skenk)	3
NAVPGSCOL	1
APL/UW, Seattle	1
ARL/PENN STATE, State College	1
Center for Naval Analyses (Acquisition Unit)	1
DDC, Alexandria	12
Marine Physical Lab, Scripps (V. Anderson)	1
MAAG, Argentina	1
MAAG, Brazil	1
SACLANT ASW Research Centre	1